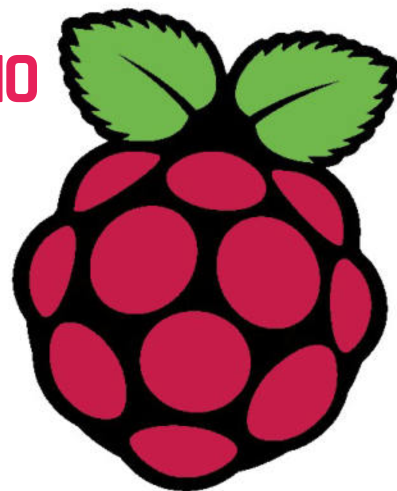
LA RIVISTA UFFICIALE RASPBERRY PI IN ITALIANO

The MagPi



Numero 77 | Gennaio 2019 | magpi.cc raspberryitaly.com

La rivista ufficiale Raspberry Pi tradotta in italiano da RaspberryItaly

20 Fantastici Accessori

I migliori kit e componenti Pi

FATTO COL CODICE

Impara Python e controlla l'elettronica con Raspberry



Estratto dal numero 77 di The MagPi. Traduzione, revisione testi e impaginazione di Mauro "Zzed" Zoia (zzed@raspberryitaly.com), per la comunità italiana Raspberry Pi www.raspberryitaly.com. Distribuito con licenza CC BY-NC-SA 3.0 .



RaspberryItaly.com COMMUNITY ITALIANA

The MagPi magazine is published by Raspberry Pi (Trading) Ltd., Mount Pleasant House, Cambridge, CB3 0RN. ISSN: 2051-9982.

FATTO COL CODICE

Se hai acquistato un Raspberry Pi per imparare a programmare e hackerare nuovi aggeggi elettronici, hai fatto una grande scelta. Inizia con questa guida ...

FARE E LANCIARE UN PROGRAMMA

Se hai appena ricevuto un nuovo e fiammante Raspberry Pi, potresti averlo già collegato e fatto funzionare. Potresti aver lanciato alcuni giochi o provato le applicazioni, o forse hai caricato uno degli strumenti per la programmazione, lo hai guardato chiedendoti cosa farci. Se non hai mai programmato prima, abbiamo, qui per te, qualcosa da sapere. Nelle prossime pagine troverai delle risposte e probabilmente anche alcune domande, ma poi ancora risposte. Prima che tu te ne renda conto, sarai diventato un ninja dell'hacking e della programmazione.

> "Se non hai mai programmato prima, tutto può sembrare complicato, ma in realtà è semplice"

Se sei nuovo della programmazione, tutto questo può sembrare un po' troppo complicato, ma in realtà è abbastanza semplice – sei solo all'inizio nel restare bloccato, e iniziare con delle cose semplici, ti porterà a dei risultati. Uno dei linguaggi di programmazione forniti con Raspberry Pi è Python. È molto facile iniziare con Python e può essere utilizzato per programmare molti dei componenti aggiuntivi che sono disponibili per Raspberry Pi – quindi, sarà molto utile impararlo. Possiamo farti programmare in 30 secondi netti; continua a leggere...

Iniziare a programmare è davvero facile. Programmare significa solo dare al Raspberry Pi un'istruzione per fare qualcosa. L'unica cosa che devi sapere è che linguaggio usare per parlarci. In questo caso, gli parleremo in Python 3. Avremo bisogno di scrivere le nostre istruzioni da qualche parte, quindi, per iniziare, apri una finestra del Terminale: fai clic sull'icona che sembra una una scatola grigia con una barra blu in cima, in alto a sinistra sullo schermo. Aprirà una finestra nera con un prompt in verde: **pi@raspberrypi:~ $**. Se si digiti `python3` e premi **INVIO**, Python si avvierà, e vedrai il prompt con il triplo simbolo di maggiore: **>>>**. Adesso scrivi `print ("Ciao")` e premi **INVIO**. Bingo! Il tuo primo programma. Hai istruito il tuo Raspberry Pi per scrivere la parola "Ciao" e, se tutto è andato bene, ha obbedito.

Immettere programmi in questo modo non sarà il modo migliore il più delle volte, andiamo quindi ora a vedere una app con cui scrivere e salvare un programma. Vai al menu del desktop (fai clic sul logo Raspberry Pi in alto a sinistra sullo schermo) e nella sezione programmazione, seleziona Thonny (che dovrebbe quindi aprirsi di default nella sua modalità semplice). Prova a digitare il seguente programma nell'editor Thonny e salvalo, quindi eseguilo facendo clic il pulsante Run. L'output verrà visualizzato nel riquadro della shell sotto l'editor del programma.



Mark Vanstone

Autore di software educativo dagli anni novanta, autore della serie ArcVenture, scomparso nel deserto delle aziende di software. Salvati dal Raspberry Pi!

magpi.cc/YiZnxL
@mindexplorers

mwc1.py

Linguaggio: **Python**
magpi.cc/HNJhhd

```
import random

correct = False
r = random.randint(1,10)
c = 0
while correct == False:
    n = input("Indovina un numero tra 1 e 10: ")
    c = c + 1
    if int(n) == r:
        correct = True
    else:
        if int(n) > r:
            print("Mi spiace, il mio numero è più basso. Riprova.")
        else:
            print("Mi spiace, il mio numero è più alto. Riprova")

print("Molto bene. La risposta corretta era " + str(r) + ".
L'hai indovinato in " + str(c) + " tentativi.")
```

IMPARA A PROGRAMMARE

La programmazione comporta l'utilizzo di diversi elementi. Vediamone qualcuno e come funzionano



▲ Dopo aver inizializzato i dati che servono al programma, creiamo un ciclo, aggiornando la variabile answer finché il giocatore non indovina tutte le lettere o termina i tentativi

Nell'ultimo programma, catturiamo l'input da tastiera e produciamo testo e numeri come output usando la funzione `print ()`. Abbiamo anche usato una struttura di condizioni, sotto forma di `if` e `else`. Python è molto particolare sull'identazione del codice con degli spazi (quattro per ogni livello di indentazione); serve a evidenziare che il codice rientrato si trova all'interno di un'altra struttura. Nell'ultimo programma, tutto ciò che è rientrato dopo il l'istruzione `while`, farà parte di quel ciclo. Diamo uno sguardo ad altre tecniche di codifica.

01 Usare le liste

Scriveremo un gioco tipo "l'impiccato", dove iniziamo con una parola segreta e il il giocatore deve indovinarla, lettera per lettera. Se corretta, mostriamo dove appare quella lettera nella parola. Sono ammessi dieci tentativi sbagliati prima della fine del gioco. Vedi il codice **mwc2.py**. Per prima cosa, creiamo un elenco di parole tra cui scegliere. In Python una lista viene definita usando parentesi quadre, tipo: `list = ["a","b","c"]`. Chiameremo la nostra lista `WORDLIST`. In questo caso stiamo scrivendo il nome della lista in maiuscolo per mostrare che è una costante, cioè che non cambierà durante il programma.

02 Scegliere una parola

Possiamo scegliere una parola dalla nostra lista usando il modulo `random`. Importiamo il modulo all'inizio del nostro codice, quindi possiamo usare la funzione `random.choice()` per ottenere una parola e memorizzarla in una variabile: `theWord`. Quando richiamiamo una funzione che è dentro un modulo, usiamo un punto tra il nome del modulo e il nome della funzione. poi, vogliamo che il giocatore inizi a tentare di indovinare qual è la parola. Se guardiamo in fondo al codice, possiamo vedere che richiamiamo una funzione chiamata `startGuessing()`. Questa è una nostra funzione che dobbiamo definire.

03 Definizione delle funzioni

Ogni volta che richiamiamo una funzione, il codice al suo interno, viene eseguito. A volte le funzioni danno dei risultati, come la nostra funzione `updateAnswer()` che restituisce la variabile `result`. Una delle regole di Python è che devi definire una funzione prima di chiamarla, quindi dovremo definire la nostra funzione `startGuessing()` nella parte superiore del codice. Per farlo, scriviamo `def` e poi il nome della funzione, seguito da parentesi e da un due punti. Se vogliamo passare delle variabili come input alla funzione, possiamo aggiungerle all'interno delle parentesi.

04 Ciclare

Ora, il codice nella nostra funzione `startGuessing()`. Impostiamo il numero di tentativi e di trattini, uno per ogni lettera della parola segreta, quindi creiamo un ciclo. Il codice dice: "Finché il giocatore ha dei tentativi e la risposta data non è la parola segreta, esegui il seguente codice". Nel nostro ciclo, stampiamo la risposta che abbiamo finora e quanti tentativi ha

Guide di programmazione

Vuoi saperne di più sulla programmazione? Dai un'occhiata alle nostre guide su Python e alla programmazione orientata agli oggetti su MagPi 53 e 54

bit.ly/2T5aMVz
bit.ly/2rT7zgu





mwc2.py

Linguaggio: Python
magpi.cc/RqQdhR

```
import random

WORDLIST = ["arancione", "tavolo", "gennaio", "pallone",
"mouse", "altoparlante", "camion"]
theWord = random.choice(WORDLIST)

def startGuessing():
    triesLeft = 10
    answer = "-" * len(theWord)

    while triesLeft > -1 and not answer == theWord:
        print("\n" + answer)
        print(str(triesLeft) + " tentativi rimanenti")
        guess = input("Indovina una lettera:")
        if len(guess) != 1:
            print("Indovina una lettera alla volta.")
        elif guess in theWord:
            print("Si, questa lettera è nella parola.")
            answer = updateAnswer(theWord, answer, guess)
        else:
            print("Mi dispiace, questa lettera non c’è.")
            triesLeft -= 1

    if triesLeft < 0:
        print("Oh no! Hai terminato i tentativi. La parola era: " + theWord)
    else:
        print("Ben fatto, hai indovinato. La parola era: " + theWord)

def updateAnswer(word, ans, guess):
    result = ""
    for i in range(len(word)):
        if word[i] == guess:
            result = result + guess
        else:
            result = result + ans[i]
    return result

print("Sto pensando una parola ....")
startGuessing()
```

Modulo
Un modulo è un altro file di codice. Può contenere funzioni, variabili, e dati.

Lista
Una lista è una collezione di variabili, in questo caso variabili stringa.

Ciclo
Un ciclo è una parte del codice che viene ripetuta una o più volte, di solito dipende se un'equazione risulta vera.

Input
Un input sono i dati che un programma riceve, in questo caso dalla tastiera.

Variable
Una variabile è un contenitore per i dati, in questo caso una stringa che può variare mentre il programma è in esecuzione.

Funzione
Una funzione contiene del codice che può essere richiamato da un altro punto del programma.

Output
L'output è qualsiasi cosa che il programma produce, in questo caso parole stampate sulla finestra di shell.

Condizione
Una condizione ramifica un programma per eseguire una parte di codice o un'altra. A volte ci sono diversi rami.

Richiamare una funzione
Quando viene richiamata una funzione, il codice all'interno della funzione viene eseguito e quindi ritorna alla linea successiva alla chiamata della funzione.

ancora il giocatore. Quindi usiamo una struttura condizionale `if`, `elif`, `else` per rispondere al giocatore, a seconda di ciò che ha digitato.

05 Cambiare la risposta

Se il giocatore indovina una lettera correttamente, noi richiamiamo un'altra funzione: `updateAnswer()`. Questa utilizza un ciclo `for` per aggiungere le lettere corrette nella nostra variabile answer, quindi restituisce tale stringa (una variabile contenente lettere/caratteri anziché un valore numerico). Questa poi diventa la variabile answer che stampiamo all'inizio di ogni ciclo nella funzione `startGuessing()`. Quando il giocatore ha indovinato tutte le lettere giuste nella nostra parola o ha sbagliato dieci volte, il programma uscirà dal ciclo per raggiungere la parte finale della funzione.

06 Vincere o perdere

Abbiamo una struttura `if` e `else` per mostrare congratulazioni o far sapere al giocatore che ha terminato i tentativi. Alcune funzioni sono usate con variabili: `len()` trova la lunghezza di una stringa, e `str()` converte una variabile numero in una stringa così può essere aggiunto all'inizio o alla fine di un altra stringa. Al termine della funzione, ritorna a dove è stato chiamata, che è la fine del programma.



CONTROLLA LE COSE CON IL CODICE

Ora che abbiamo preso la mano con la programmazione, usiamola per far funzionare alcuni componenti elettronici

Nell'esempio precedente, abbiamo importato un modulo nel nostro codice e usato le funzioni in esso contenute. Se vogliamo controllare l'elettronica dal codice, c'è disponibile un modulo molto utile chiamato `gpiozero`. GPIO sta per "general-purpose input/output" (input/output generici) e i doppi piedini in fila su un lato del Raspberry Pi sono chiamati pin GPIO. Per dettagli sule funzioni di ogni pin, vedi **pinout.xyz**. Se colleghiamo i componenti elettronici a questi pin GPIO, possiamo usare il modulo `gpiozero` per far funzionare le cose. Quando importiamo un modulo, spesso ci sono al suo interno molte funzioni diverse. Possiamo anche usare con loro una nuova programmazione a 'oggetti'. Gli oggetti sono come variabili ma hanno il proprio set di funzioni e proprietà al loro interno che possiamo richiamare o cambiare, e lo facciamo usando la stessa notazione dot (punto) che abbiamo usato con il modulo `random`. Per maggiori dettagli sull'uso della programmazione a oggetti (chiamata programmazione orientata agli oggetti o OOP), vedi il numero 54 di The MagPi: **bit.ly/2rT7zgu**.

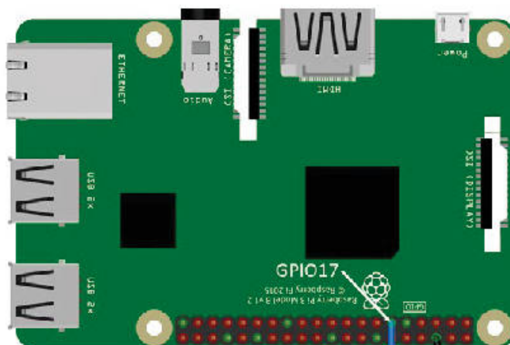
Cosa Serve

Una breadboard
magpi.cc/NtjSiy

Un LED
magpi.cc/WBVPxG

Una restistenza
magpi.cc/iDTFag

2 cavallotti maschio - femmina
magpi.cc/OkMyVX



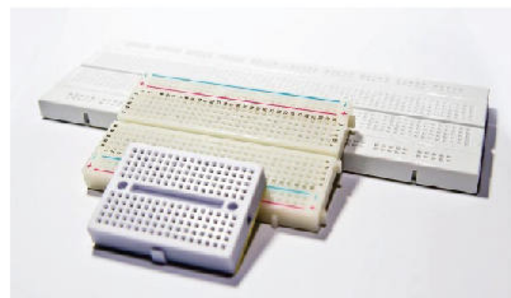


Questa resistenza è da 1 kΩ (kilohm), ma puoi usare un valore simile

Questo è un LED giallo, ma puoi comprare LED di molti colori diversi

01 La breadboard

Le breadboard sono disponibili in varie dimensioni, ma funzionano tutte allo stesso modo. Se vi sono segnate piste in rosso e nero/blu (e/o + e –) sui bordi lunghi della tavoletta, queste sono per l'alimentazione e sono collegate lungo la loro lunghezza – anche se a volte divise in sezioni. La matrice di fori che compongono la parte principale della tavoletta è collegata in file nell'altro senso (verticalmente nel diagramma). Di solito c'è un separatore al centro del tabellone in modo che i due lati non siano collegati.

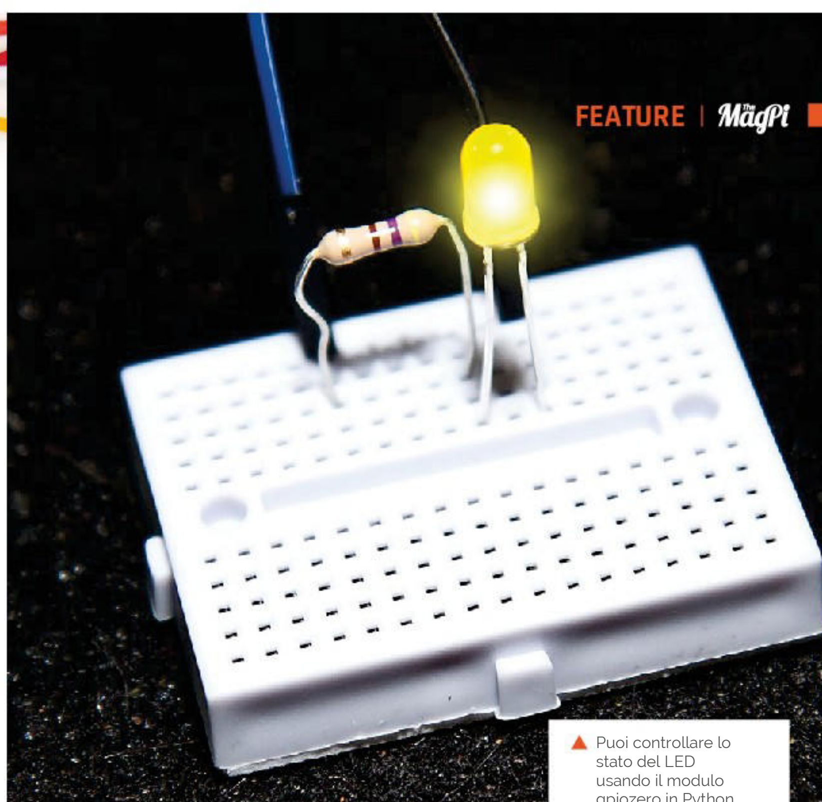


02 Diodi emettitori di luce (LED)

Un LED è un po' come una lampadina, se si applica elettricità nel modo giusto, si accende. Un LED è anche un diodo, il che significa che l'elettricità ha bisogno di fluire nella direzione corretta o non lo farà accendere. Quando colleghiamo un LED al Raspberry Pi, dobbiamo

aggiungere una resistenza al circuito, poiché la maggior parte dei LED brucerà se li colleghiamo direttamente all'alimentazione. Nell'esempio, stiamo usando una resistenza da 1 kΩ, ma si può usare un altro valore simile. Per accendere un LED usando `gpiozero`, assembla i componenti come nello schema, quindi scrivi un programma Python: comincia con `from gpiozero import LED`, quindi crea un oggetto LED con `led = LED (17)` e infine digita `led.on()` per accendere il LED.





▲ Puoi controllare lo stato del LED usando il modulo gpiozero in Python

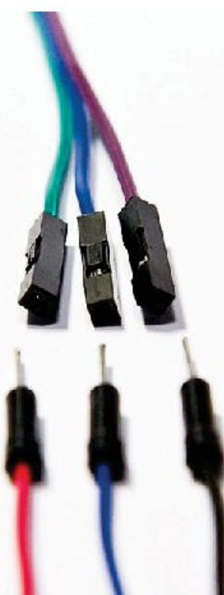
03 Resistenza

Le resistenze fanno quello che suggerisce il loro nome: resistono al flusso di elettricità (corrente). Alcuni componenti hanno bisogno di avere una certa quantità di corrente per funzionare correttamente. Le resistenze ci permettono di regolare la corrente o la tensione a un livello adatto per gli altri componenti che stiamo usando. Ci sono molti diversi tipi di resistenza, con diversi valori. Il valore della resistenza può essere letto dal codice a strisce di colore sul suo corpo. Puoi avere anche resistenze variabili, che sono note anche come trimmer o potenziometri.



04 Cavallotti

Dobbiamo collegare i nostri componenti ai pin GPIO del Raspberry Pi. Per farlo, usiamo questi fili a ponticello, o cavallotti. Quelli che vogliamo utilizzare hanno un connettore femmina ad una estremità, per inserirsi sui pin GPIO, e un connettore maschio per inserirsi nei fori della breadboard. Esistono anche cavallotti con entrambi i connettori maschi o entrambi i connettori femmina, per diverse situazioni. Possono essere acquistati a strisce, tutti uniti insieme, a volte chiamate 'Jumper jerky'.



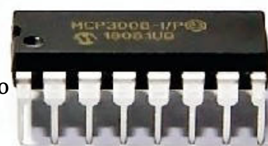
05 Potenziometro

Un potenziometro è una resistenza variabile. Di solito ha un perno rotante che modifica la resistenza da un valore all'altro, quasi sempre da nessuna resistenza a resistenza massima (nessuna elettricità che scorre). Un potenziometro può fornirci una tensione in uscita variabile che possiamo misurare con il GPIO, ma c'è un piccolo problema. Il potenziometro fornisce una uscita 'analogica' (il valore può essere uno qualsiasi tra il minimo e il massimo) e il GPIO ha solo input digitali, vale a dire on o off. Quindi abbiamo bisogno di un altro componente: un convertitore analogico-digitale (ADC).



06 Convertitore analogico-digitale

I componenti ADC sono disponibili in varie forme, ma quello che abbiamo in questo esempio è chiamato MCP3008. È un circuito integrato (IC), che significa che è un piccolo scatolino con alcuni circuiti al suo interno. Non ci serve sapere cosa c'è dentro – abbiamo solo bisogno di sapere cosa collegare a ciascuna delle gambette del circuito integrato. Dovremo collegare molti dei piedini ai pin GPIO e dare alimentazione all'IC; quando abbiamo fatto questo, possiamo collegare il potenziometro all'IC e quindi leggere i valori, mostrando la posizione del perno del potenziometro usando il modulo `gpiozero`. Analizzeremo il codice nella prossima sezione.



Guide di elettronica

Per ulteriori dettagli sull'uso dei componenti elettronici con il Raspberry Pi, vedi la nostra guida di partenza in elettronica sulla traduzione del numero 64 di *The MagPi*.

bit.ly/2EWaLQL



PONG CON I POTENZIOMETRI

Ora mettiamo alla prova ciò che abbiamo imparato: creeremo un gioco retrò e lo controlleremo con i nostri controller fatti in casa

Cosa Serve

Breadboard
magpi.cc/vhZkGh

6 cavallotti maschio-femmina e
10 cavallotti maschio-maschio
magpi.cc/ffZNwL

2 potenziometri
magpi.cc/oZRFEe

Circuito integrato MCP3008
magpi.cc/aCvXuo

Nella pagina precedente, abbiamo parlato di potenziometri e convertitori analogico-digitali, e qui è dove possiamo usarli. È un poco più complicato che accendere un LED, ma solo un po'. Primo, andremo a scrivere un programma che ha due racchette rettangolari su entrambi i lati dello schermo che possono essere spostate su e giù da due giocatori. Una palla rimbalza avanti e indietro tra le due racchette, fino a quando un giocatore perde la palla e l'altro giocatore segna un punto. Esatto, hai indovinato: il gioco è Pong e stiamo creando un controller per ogni giocatore, con un potenziometro che collegheremo al Raspberry Pi.

01 Programmazione rapida

Se hai seguito gli altri articoli sulla programmazione nei recenti numeri di *The MagPi*, saprai che quando si scrive un gioco veloce sul Raspberry Pi, Pygame Zero è tuo amico. Possiamo realizzare le basi del codice del gioco molto rapidamente importando il modulo `pgzrun`, che contiene tutto il codice Pygame Zero. Dobbiamo richiamare `pgzrun.go()` alla fine del nostro codice, e questa è la nostra finestra di gioco in ordine. Come con tutti i programmi Pygame Zero, abbiamo una funzione `draw()` per disegnare la grafica nella finestra, e una funzione `update()` per aggiornare gli elementi del gioco tra i cicli di disegno.

02 Lanciare il codice

Il listato **mwc3.py** fornisce tutto il codice che ti serve affinché che il gioco funzioni. C'è del codice nella funzione `updatePaddles()` per controllare i paddle o racchette da tastiera, nel caso volessi testarlo prima di creare i controller personalizzati. Importiamo diversi moduli con questo codice. Abbiamo parlato di `pgzrun`, ma abbiamo anche bisogno di `random` in modo che la palla si muova in una direzione casuale ogni volta che il gioco Inizia. Inoltre, abbiamo bisogno di `gpiozero` per gestire l'input dei controller e ci serve il modulo matematico `math` per il calcolo della direzione della palla.

03 Cablare tutto

Una cosa da tenere a mente quando si connette un qualsiasi componente elettronico a un computer è che se i fili sono collegati nel modo sbagliato, puoi causare danni al computer o ai



Possono sembrare un sacco di fili, ma segui metodicamente lo schema per essere certo che i collegamenti siano corretti

componenti elettronici, quindi è sempre una buona idea spegnere il Raspberry Pi prima di collegare qualsiasi cosa ai pin GPIO. Segui attentamente lo schema elettrico (sul retro), assicurandoti che i cavi dei ponticelli siano collegati ai pin GPIO corretti e nei posti giusti sulla breadboard. Una volta che hai fatto tutto nel modo corretto, è una buona idea fare un ulteriore controllo, giusto per sicurezza.

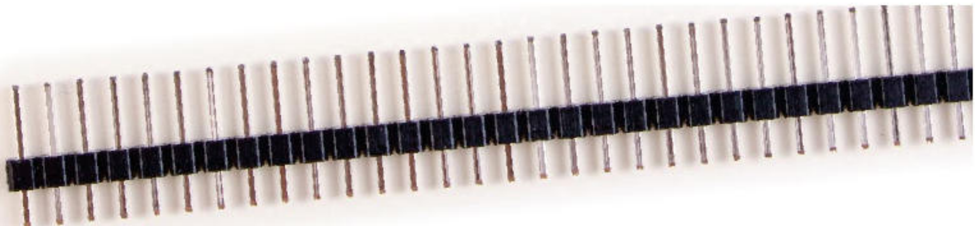
04 L' IC MCP3008

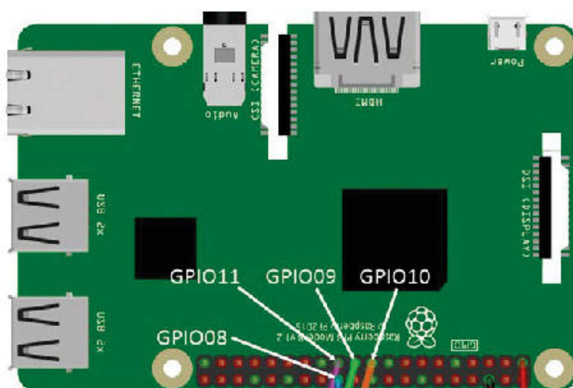
L'MCP3008 converte la tensione dai nostri potenziometri in un numero, con l'aiuto del modulo gpiozero. Ha otto canali per l'input, ma in questo caso ne useremo solo due. Come vedi dallo schema, che tutte le gambette sul lato superiore dell'IC sono collegate ai pin GPIO o alle linee di alimentazione. Ci sono due collegamenti segnati in rosso che vanno alla linea di alimentazione positiva, poi uno nero che porta alla linea negativa o massa. Poi ci sono quattro fili colorati così collegati: viola - GPIO11; verde - GPIO09; arancione - GPIO10; e blu - GPIO08. Infine, c'è un ultima connessione alla linea di massa.

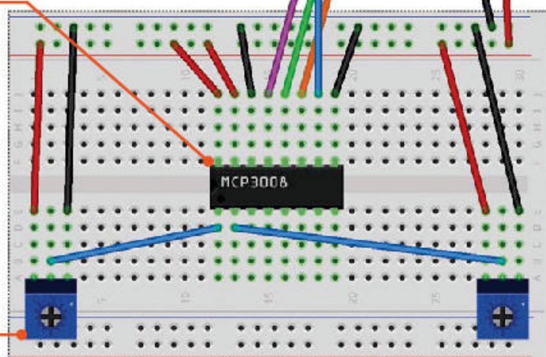
05 Gli ingressi

Tutti i pin dell'MCP3008 sul lato inferiore dell'IC sono canali di input. Useremo i primi due pin, che sono il canale 0 e il canale 1.









Il MCP3008 è a cavallo della scanalatura al centro della breadboard così che i piedini di entrambi i lati non risultino collegati tra loro

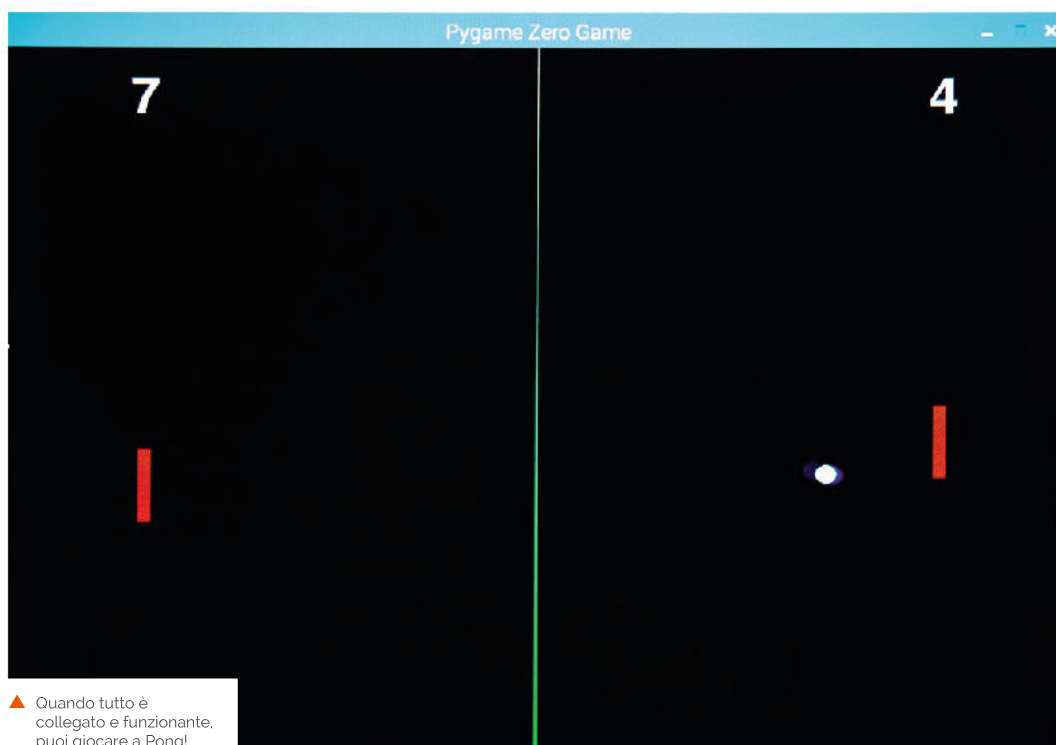
Il pin centrale del potenziometro di sinistra si connette al pin 0 dell'MCP3008 quello destro al pin 1

Colleghiamo i pin centrali dei nostri potenziometri a questi pin dei canali, che leggeranno le posizioni e le convertiranno in un valore nel nostro programma. Se vuoi sapere a cosa servono tutti i pin di questo IC, puoi fare una ricerca sul web con 'MCP3008 pinout 'e otterrai la descrizione di ciascun piedino. I circuiti integrati sono molto utili nell'elettronica in quanto consentono di ridurre la complessità dei nostri circuiti e non è necessario che noi sappiamo esattamente come funzionano al loro interno. Sono un po' come i moduli in Python.

06 Finire il lavoro

Quando hai controllato che tutte le connessioni sono corrette, puoi accendere il Raspberry Pi e ricaricare il tuo programma. Potresti voler abilitare l'interfaccia SPI sul tuo Raspberry Pi andando nel menu principale del desktop > Preferenze> Configurazione Raspberry Pi, e poi nella scheda Interfacce. Funzionerà anche senza, ma potrebbe causare alcuni avvertimenti nella finestra shell di Thonny. Quindi, se è tutto giusto, quando lanci il tuo programma avrai un gioco di Pong che può essere controllato da due giocatori con i potenziometri. Se non funziona la prima volta, controlla il codice e poi il tuo cablaggio, e riprova.





▲ Quando tutto è collegato e funzionante, puoi giocare a Pong!

mwc3.py

Linguaggio: Python
magpi.cc/umUcfq

```python
import pgzrun
import random
from gpiozero import MCP3008
import math

pot1 = MCP3008(0)
pot2 = MCP3008(1)

# Set up the colours
BLACK = (0  ,0  ,0  )
WHITE = (255,255,255)
p1Score = p2Score = 0
BALLSPEED = 5
p1Y = 300
p2Y = 300

def draw():
    screen.fill(BLACK)
    screen.draw.line((400,0),(400,600),"green")
    drawPaddles()
    drawBall()
    screen.draw.text(str(p1Score) , center=(105, 40), color=WHITE, fontsize=60)
    screen.draw.text(str(p2Score) , center=(705, 40), color=WHITE, fontsize=60)

def update():
    updatePaddles()
    updateBall()

def init():
    global ballX, ballY, ballDirX, ballDirY
    ballX = 400
    ballY = 300
    a = random.randint(10, 350)
    while (a > 80 and a < 100) or (a > 260 and a < 280):
        a = random.randint(10, 350)
    ballDirX = math.cos(math.radians(a))
    ballDirY = math.sin(math.radians(a))

def drawPaddles():
    global p1Y, p2Y
    p1rect = Rect((100, p1Y-30), (10, 60))
    p2rect = Rect((700, p2Y-30), (10, 60))
    screen.draw.filled_rect(p1rect, "red")
    screen.draw.filled_rect(p2rect, "red")

def updatePaddles():
    global p1Y, p2Y

    p1Y = (pot1.value * 540) +30
    p2Y = (pot2.value * 540) +30

    if keyboard.up:
        if p2Y > 30:
            p2Y -= 2
    if keyboard.down:
        if p2Y < 570:
            p2Y += 2
    if keyboard.w:
        if p1Y > 30:
            p1Y -= 2
    if keyboard.s:
        if p1Y < 570:
            p1Y += 2

def updateBall():
    global ballX, ballY, ballDirX, ballDirY, p1Score, p2Score
    ballX += ballDirX*BALLSPEED
    ballY += ballDirY*BALLSPEED
    ballRect = Rect((ballX-4,ballY-4),(8,8))
    p1rect = Rect((100, p1Y-30), (10, 60))
    p2rect = Rect((700, p2Y-30), (10, 60))
    if checkCollide(ballRect, p1rect) or checkCollide(ballRect, p2rect):
        ballDirX *= -1
    if ballY < 4 or ballY > 596:
        ballDirY *= -1
    if ballX < 0:
        p2Score += 1
        init()
    if ballX > 800:
        p1Score += 1
        init()


def checkCollide(r1,r2):
    return (
        r1.x < r2.x + r2.w and
        r1.y < r2.y + r2.h and
        r1.x + r1.w > r2.x and
        r1.y + r1.h > r2.y
    )

def drawBall():
    screen.draw.filled_circle((ballX, ballY), 8, "white")
    pass

init()
pgzrun.go()
```

I MIGLIORI ACCESSORI RASPBERRY PI

La nostra guida ai migliori add-on, HAT e espansioni per il tuo Raspberry Pi

Il Raspberry Pi è davvero fantastico. Con LAN wireless, Bluetooth, Porte USB, i versatili pin del GPIO e un sistema operativo completo, è davvero un computer perfettamente funzionante.

E se potessi migliorarlo? Fare di più? Rendere più semplici certe funzioni? Qui è dove subentrano gli add-on e gli accessori per Pi - ci sono centinaia di espansioni che puoiaggiungere al tuo Raspberry Pi che faranno migliorare le sue caratteristiche già impressionanti.

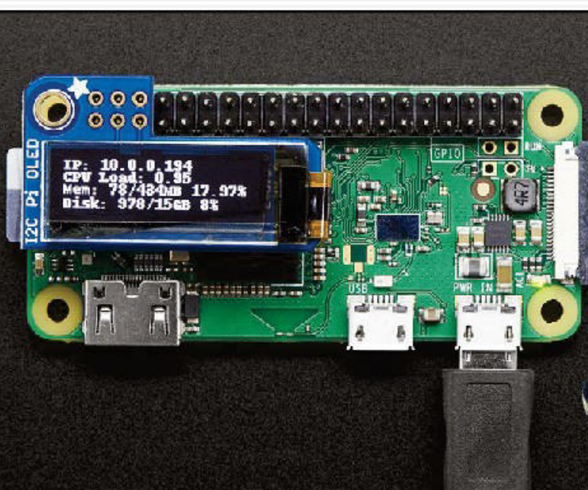
Quali componenti aggiuntivi dovresti voler ottenere, però? Siamo qui per mostrarti il meglio che il mondo Raspberry Pi ha da offrire.

Tipologie di accessori Pi

Il glossario dei termini **add-on**

HAT

Inizieremo con uno importante.
HAT sta per 'hardware in testa – hardware attached on top'(è un acronimo inverso), ed è uno standard per gli add-ons. Ciò significa che esistono definizioni specifiche di come un HAT può connettersi e funzionare. Tutto ciò che devi pensare, con un HAT, è che si adatti perfettamente a un Raspberry Pi a dimensione intera, collegandolo sui pin GPIO. Di solito è necessario installare del software aggiuntivo per farli funzionare.

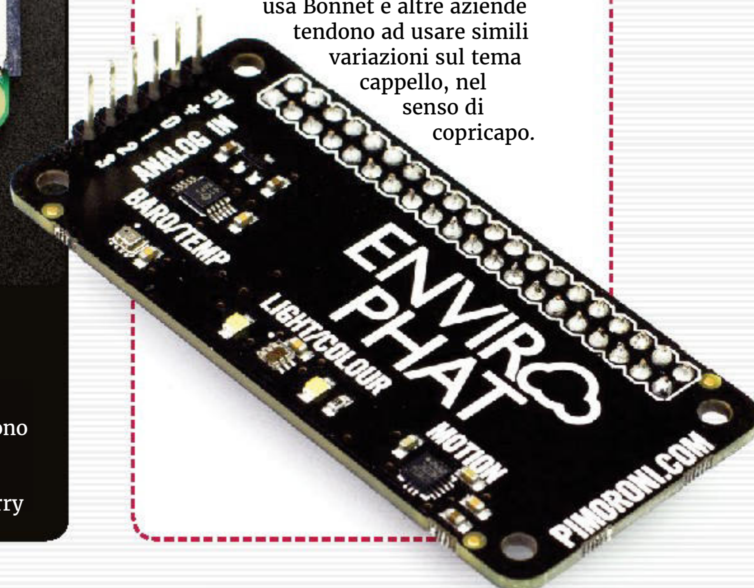


Add-on

Un HAT è un add-on, un componente aggiuntivo, ma un add-on non è necessariamente un HAT. Gli add-on possono essere collegati tramite USB, con solo alcuni specifici pin GPIO, alla porta della videocamera, ecc. È un termine generico per qualcosa che puoi aggiungere al tuo Raspberry Pi in qualche modo.

pHAT/Bonnet/Cap/ecc.

Mentre lo standard HAT è recentemente stato esteso a diverse dimensioni, molte aziende hanno creato i propri nomi per gli add-on che funzionano come un HAT ma non ne rispettano i criteri tecnici. Pimoroni ha coniato pHAT e Adafruit usa Bonnet e altre aziende tendono ad usare simili variazioni sul tema cappello, nel senso di copricapo.



PROGRAMMARE E FARE COSE CON GLI ACCESSORI

Migliora i tuoi progetti con questi add-on

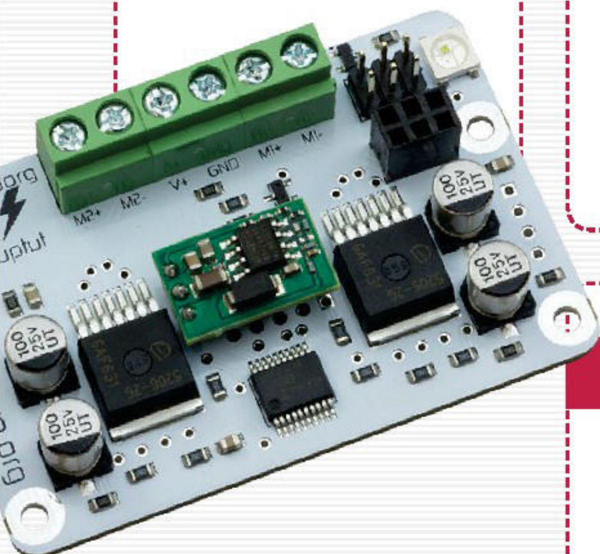
ThunderBorg

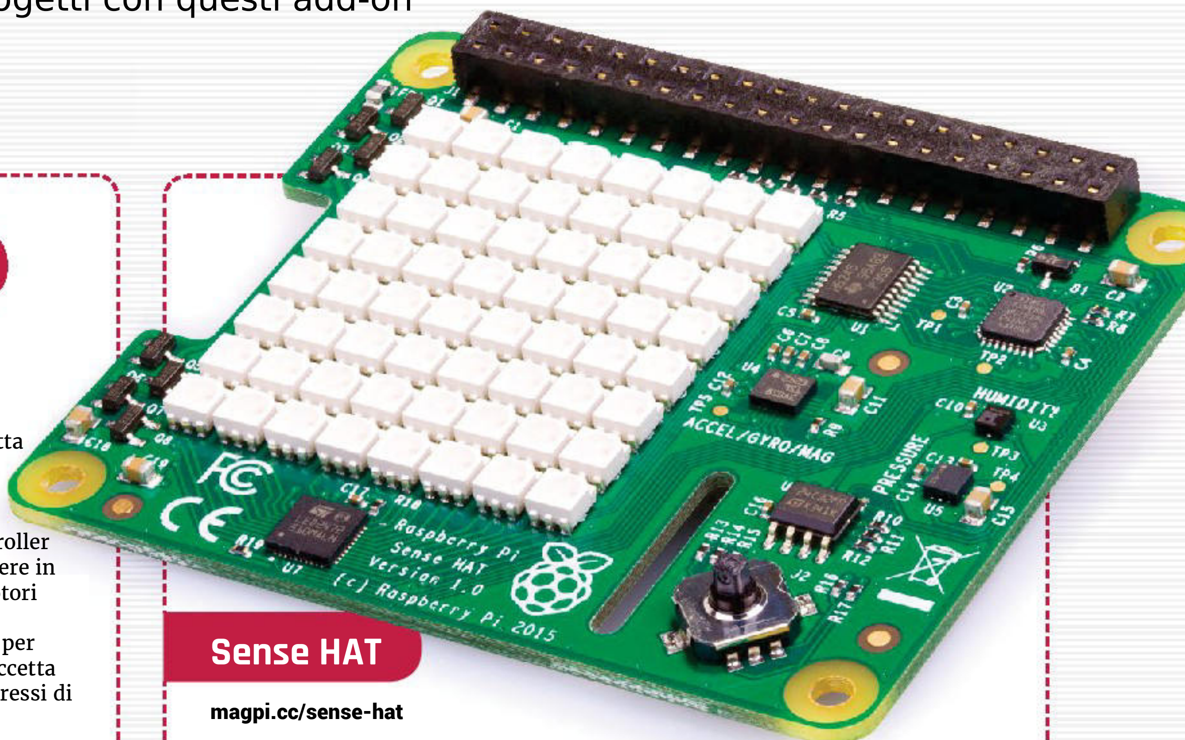
magpi.cc/xcsSpm

prezzo: 32€ / 37$

Questa scheda a doppio controller per motori fatta dai ragazzi di PiBorg è un potente e robusto modo di usare un Raspberry Pi come controller di un robot. Oltre ad essere in grado di far andare i motori avanti e indietro, può controllarne la velocità, per manovre più precise e accetta una vasta gamma di ingressi di alimentazione.

Usalo con: Robot, auto RC





Sense HAT

magpi.cc/sense-hat

prezzo: 35€ / 40$

Sviluppato per la missione Astro Pi, il Sense HAT è principalmente un insieme di sensori ambientali. Può rilevare pressione, umidità, temperatura e orientamento/movimento. Ha anche una matrice a LED 8×8 e un piccolo joystick per un display e controllo molto semplici. Al momento, ci sono un paio di Sense HAT in funzione su schede Raspberry Pi sulla ISS.

Usalo con: Stazioni meteorologiche, stazioni spaziali



Hologram Nova

hologram.io/nova

prezzo: Da 55€ / 64$

Nonostante i migliori sforzi mondiali, il WiFi non è assolutamente ancora disponibile ovunque. Non puoi nemmeno collegare un cavo Ethernet in un albero! Nova risolve questo problema permettendoti di connetterti tramite le reti cellulari di tutto il mondo con il tuo Pi tramite questo semplice dispositivo USB.

Usalo con: Progetti portatili, telecamere esterne

Edge TPU Accelerator

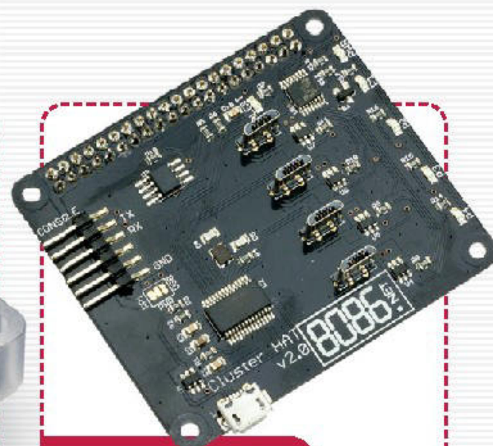
magpi.cc/TFLQpM

prezzo: Da definire

Google ha realizzato l'AIY Voice Kit che abbiamo regalato con il numero 40 della rivista *The MagPi* - uno speciale HAT e altro che ti permette di trasformare il tuo Pi in un assistente con AI. L'Edge TPU accelerator mira ad aiutare il Pi con compiti di intelligenza artificiale e rendere i tuoi assistenti vocali migliori.

Usalo con: Controllo vocale, progetti di intelligenza artificiale





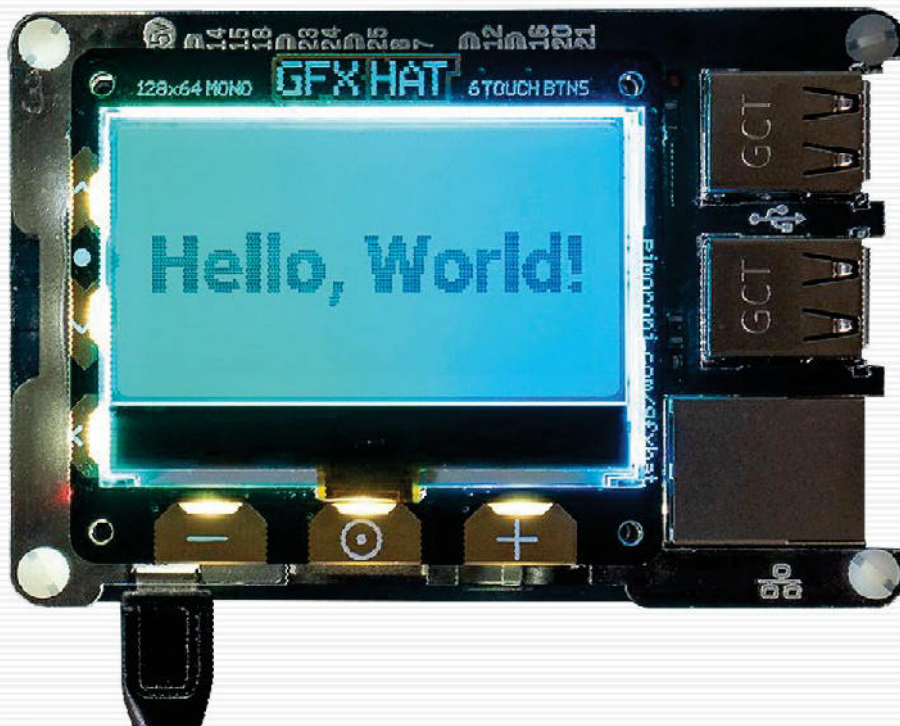
Cluster HAT

magpi.cc/Qrshbi

prezzo: 32€ / 37$

Mettere in cluster schede Raspberry Pi non è troppo difficile, ma il Cluster HAT lo rende davvero molto semplice. Usando un Raspberry Pi, puoi distribuire il calcolo fino a quattro Pi Zero. Il cluster computing è un ottimo modo per realizzare in modo diverso dei calcoli.

Usalo con: Matematica, grandi algoritmi

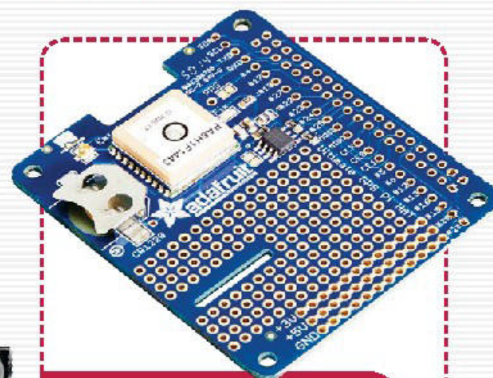


GFX HAT

magpi.cc/ZWvcLG **prezzo:** 22€ / 25$

Uno schermo molto funzionale che consuma molto meno energia rispetto alla maggior parte degli altri schermi, include anche sei pulsanti tattili capacitivi per controllarlo. È il successore di un eccellente add-on conosciuto come Display-O-Tron e rende i progetti interattivi di bell'aspetto.

Usalo con: Termostati, progetti pratici



Ultimate GPS HAT

magpi.cc/vmPqLL

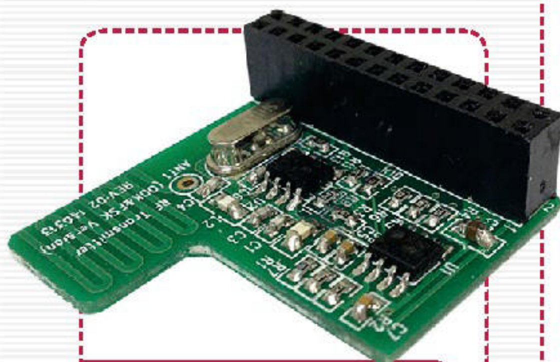
prezzo: 32€ / 37$

Questo add-on GPS consente di tracciare la posizione precisa del tuo Raspberry Pi - beh, per quanto preciso può essere il GPS. Dovrai combinare i dati di posizione con dati di mappa per farne un grande uso, ma una volta che hai tutto funzionante è fantastico.

Usalo con: Navigatori satellitari, tracciamento delle rotte

POTENZA INDUSTRIALE

Utilizzi un Raspberry Pi a livello industriale? Potresti volere questi...

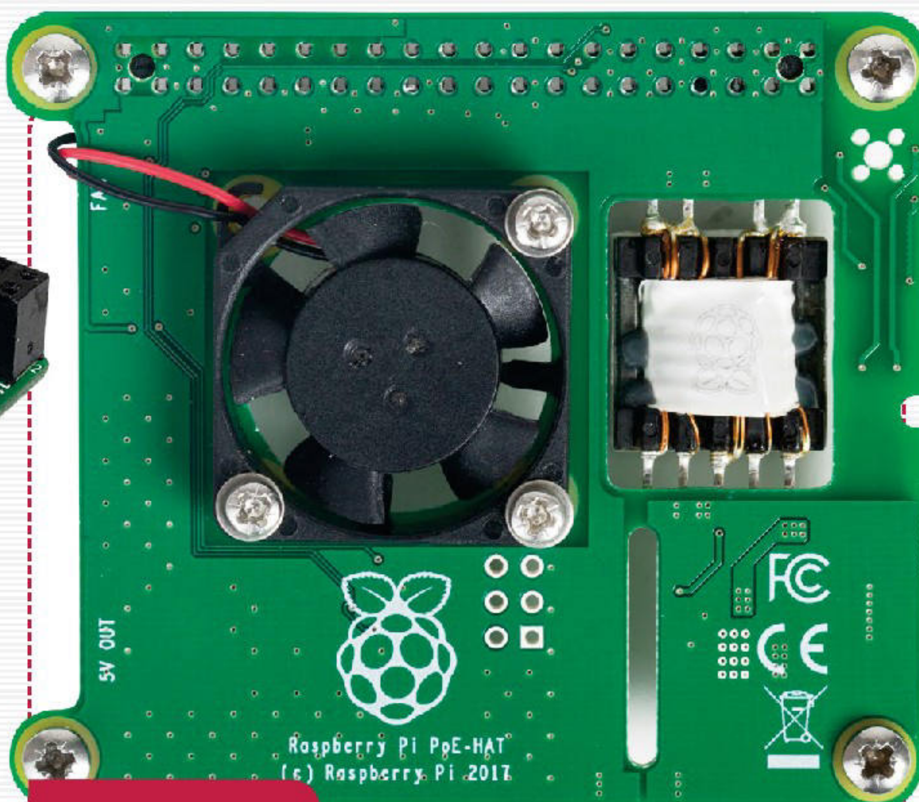


Energenie Pi-mote

magpi.cc/FnezCP
prezzo: 24€ / $28

Un'altra soluzione di controllo remoto di apparecchi, è stato progettato per il Pi per controllare l'alimentazione a distanza. Ci sono vari prodotti Energenie Pi-mote, il più economico parte da 11€, ma anche la versione del kit di base include prese compatibili. c'è anche un Pi-mote che consente alle prese di parlare con il Pi.

Usalo con: illuminazione a tempo, alcune applicazioni IoT

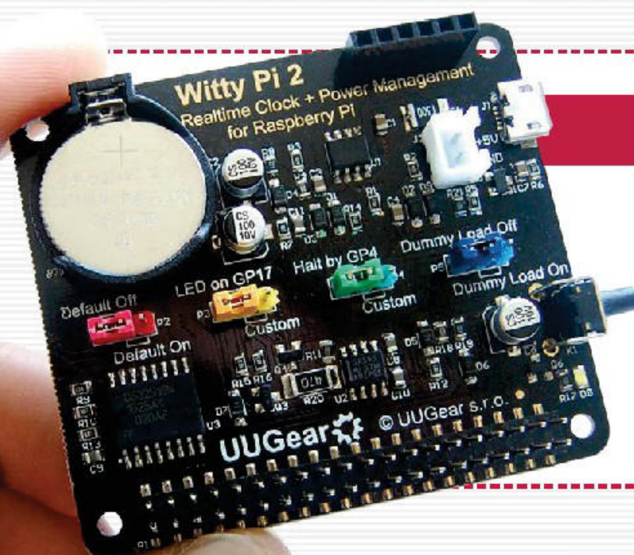


PoE HAT

magpi.cc/aqpwZc **prezzo:** 20€ / $21

Il Raspberry Pi 3B+ è il primo modello Pi ad includere un adeguato supporto al Power-over-Ethernet, anche se è necessario uno di questi HAT per farlo funzionare. Ovviamente, la rete deve essere configurata per supportare (alimentare) sia il PoE che il Raspberry Pi. Attualmente funziona solo su Raspberry Pi 3B+.

Usalo con: Server rack, soluzioni headless

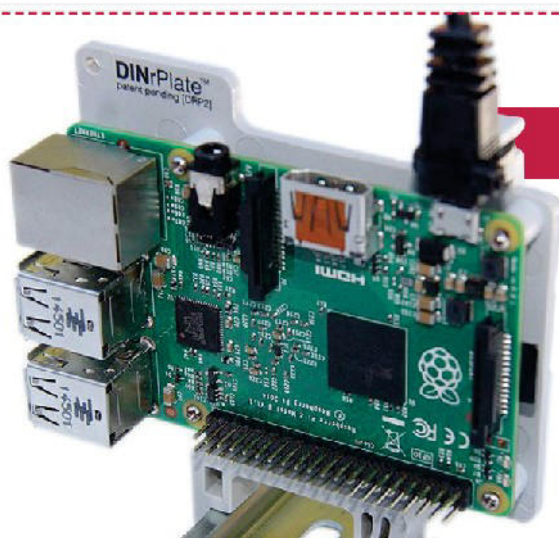


Witty Pi 2

magpi.cc/merPpf
prezzo: 22€ / $26

Gli utenti seri di Raspberry Pi spesso hanno bisogno di un sistema di gestione dell'alimentazione e di un orologio in tempo reale (RTC) per vari motivi. Il Witty Pi 2 offre queste cose, insieme ad alcune funzioni extra che possono essere programmate (on/off). Siccome Raspberry Pi normalmente richiede Internet per controllare l'ora, un RTC è essenziale per pianificare attività su un Pi che non è online.

Usalo con: Settaggi industriali, soluzioni automatizzate



DINrPlate Montaggio su DIN

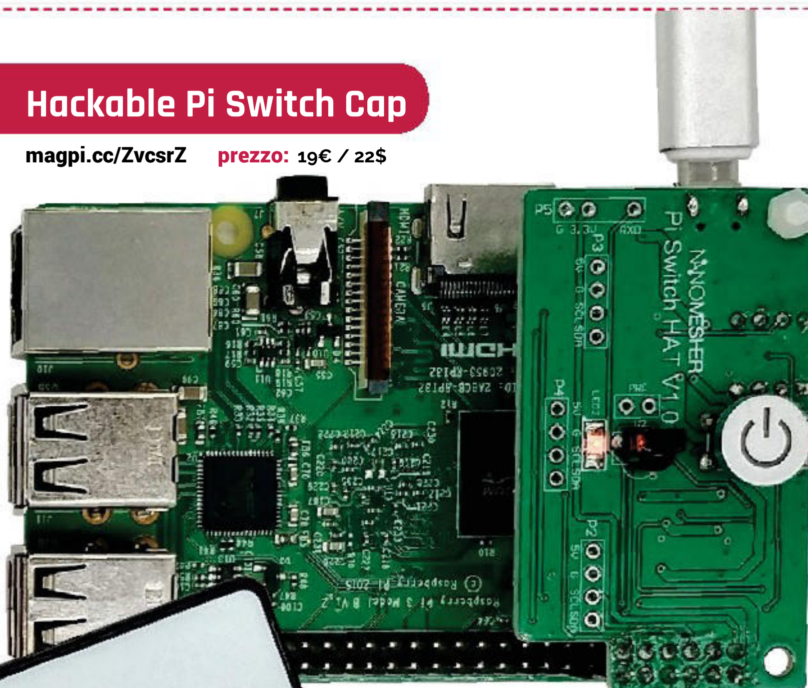
dinrplate.com
prezzo: 11€ / 13$

Vuoi utilizzare un Raspberry Pi in un ambiente server, ma non hai bisogno di un UPS o RTC o di qualcosa di simile? hai comunque qualche guida DIN a portata di mano? Allora il DINrPlate è un eccellente add-on che ti permette di montare un Pi su una guida DIN senza altri ammennicoli. E il prezzo riflette questa filosofia.

Usalo con: Guide DIN, server fai-da-te

Hackable Pi Switch Cap

magpi.cc/ZvcsrZ **prezzo:** 19€ / 22$



Soluzioni per interruttore on/off per Raspberry Pi, esistono da quando esiste il Pi stesso, ma mai nessuno di loro è stato così bello. Arriva l'Hackable Pi Switch Cap di Nanomesher e finalmente ne abbiamo una che fa tutto il necessario. Di default, ti permette di accendere e di spegnere il Pi con un pulsante, e puoi riprogrammalo per fare di più con il telecomando incluso.

Usalo con: Pi Desktop, HTPC

Strato Pi CM

magpi.cc/rfBfvo
prezzo: 222€ / 256$

Sei amministratore di sistema in una grande società che vorrebbe utilizzare schede Raspberry Pi? Lo Strato Pi trasforma un Pi in un server abbastanza buono per passare i rigorosi test di IT aziendale. Ha un impressionante elenco di funzionalità: RTC, UPS, custodia per guida DIN a due moduli, RS-485 e altro.

Usalo con: Enterprise, server



ADD-ON MULTIMEDIALI

Migliora la **visione**, l'**ascolto** e **altro**

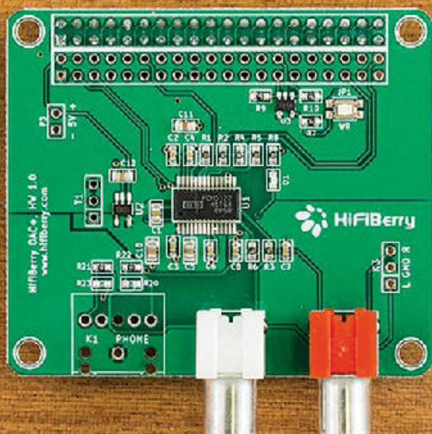
HiFiBerry DAC+

magpi.cc/aimAyy

prezzo: 26€ / $29

Esistono molti convertitori da digitale ad analogico (DAC) che trasformano il tuo Pi in un potente stereo, ma noi tendiamo alla gamma HiFiBerry DAC+. Ci sono versioni più economiche, versioni più piccole, versioni più potenti e tutte consentono che il Pi suoni un po' meglio del normale.

Usalo con: Sistema stereo, PA





HyperPixel 4.0

magpi.cc/WgSmAA

prezzo: 44€ / $50

L'HyperPixel 4.0 è uno schermo enorme e meraviglioso per il Pi che puoi avere con o senza la funzionalità touch capacitiva. È semplice da configurare e, grazie all'alta densità di pixel, è ottimo per guardare i media o le foto.

Usalo con: Progetti multimediali portatili, schermi a muro



USB SNES controller

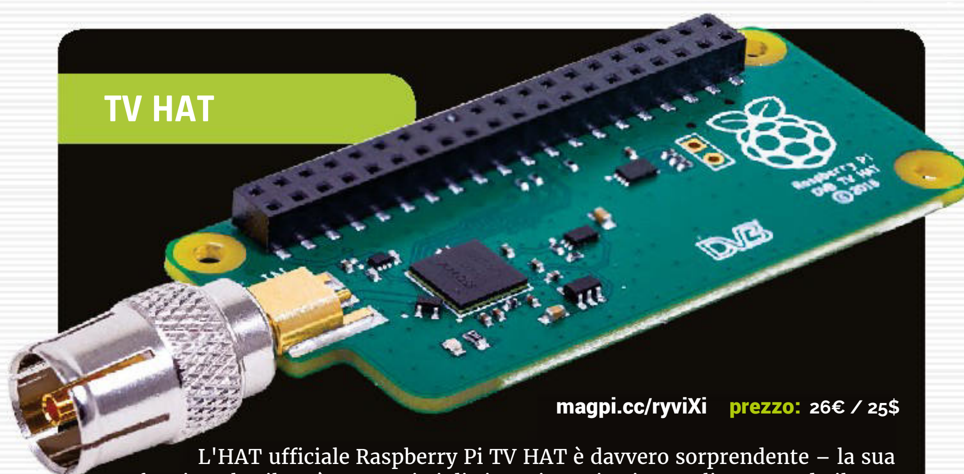
magpi.cc/2ve6iUo

prezzo: 7€ / $8

Se stai giocando a giochi retrò con il tuo Raspberry Pi, potresti volere un controller retrò per farlo sembrare più autentico. Noi consigliamo una delle tante versioni del controller SNES classico, che abbia abbastanza pulsanti per giocare a qualsiasi gioco sia prima del 1996 che successivo.

Usalo con: Retro console, progetti di veicoli RC

TV HAT



magpi.cc/ryviXi **prezzo:** 26€ / 25$

L'HAT ufficiale Raspberry Pi TV HAT è davvero sorprendente – la sua funzione basilare è consentirti di sintonizzarti sui segnali TV usando il tuo Pi. Quello è l'uso standard, ma la parte sorprendente è la sua capacità di trasmettere in streaming questa TV non solo lungo la tua rete, ma anche su Internet se sei all'estero e ti manca l'ultimo episodio della tua trasmissione preferita.

Usalo con: DVR fai-da-te, TV ovunque tu sia



FLIRC USB

flirc.tv

prezzo: 22€ / 23$

I telecomandi hanno alcuni buoni standard al giorno d'oggi, e per cose come Kodi ci sono molti modi per controllare il tuo HTPC basato su Pi, tramite il tuo telefono o tablet. FLIRC rende il tutto un po' più semplice, però: prendi il tuo telecomando preferito, dì a FLIRC quale pulsante fa cosa, e quindi collegalo al tuo HTPC. Semplice.

Usalo con: Kodi, IoT telecomandata

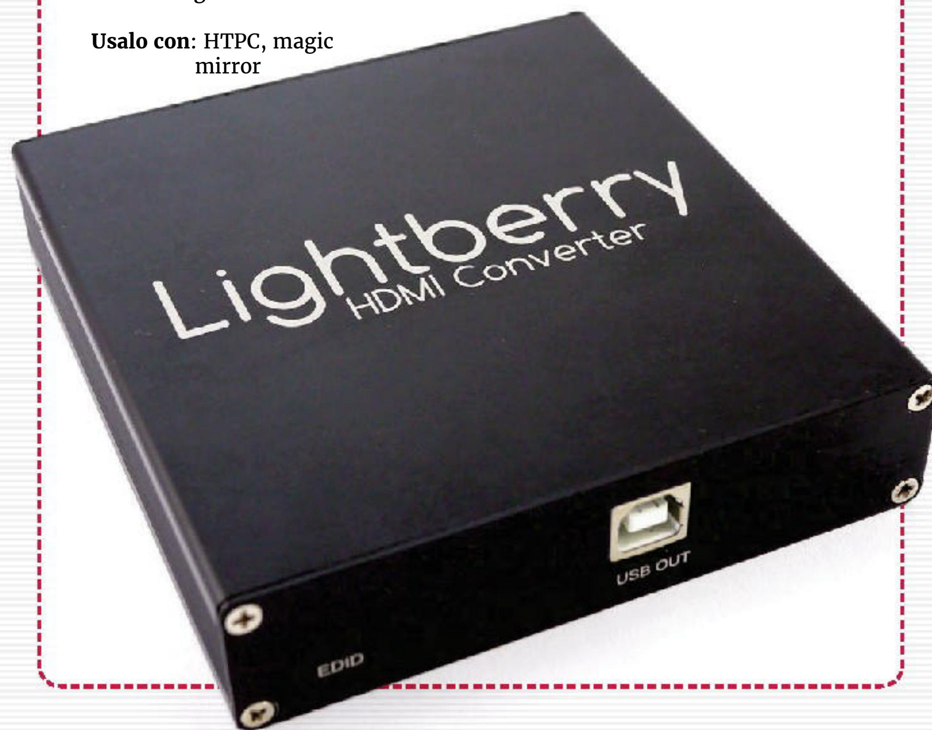
Lightberry HD

lightberry.eu

prezzo: 106€ / 120$

Hai mai desiderato che i tuoi film si espandessero dalla tua TV? Con Lightberry, puoi installare effetti fai-da-te tipo Ambilight sulla tua TV in modo che gli effetti speciali diventino un po' più speciali. Ci sono vari kit tra cui scegliere.

Usalo con: HTPC, magic mirror





Picade X HAT

magpi.cc/BupAFF

prezzo: 17€

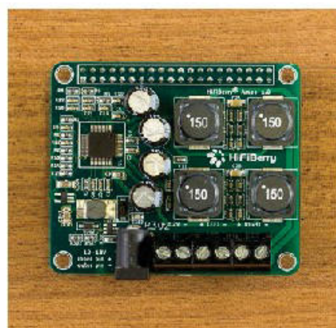
The Picade è un eccellente kit da tavolo arcade di Pimoroni, e la sua ultima iterazione è controllata dai potenti Picade X HAT. È così ben fatto, che hanno reso possibile comprarlo separatamente per usarlo nella propria configurazione arcade fai-da-te.

Usalo con: Costruzione di arcade, telecomandi

I 10 Migliori DAC

Migliora la qualità del suono del tuo Raspberry Pi con questi convertitori digitali-analogici

Non si vuole screditare il Raspberry Pi – dopotutto la sua uscita audio non suona mica come una logora videocassetta nel 1992. Puoi ottenere del buon suono standard da esso - tuttavia, si può fare ancora molto meglio. In questa nicchia si inserisce una intera schiera incredibili accessori DAC.



HiFiBerry Amp+

Andando oltre il DAC, c'è questo amplificatore che puoi utilizzare come parte di un sistema stereo serio. È di classe D e tutto quel che devi fare è connettere i tuoi altoparlanti. È fantastico per installazioni multiroom.

- 50€ / 57$
- magpi.cc/KWNSpN

Nanosound DAC

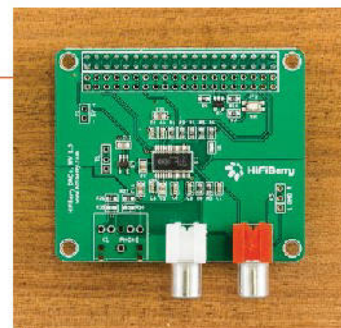
Questo potente DAC è disponibile in una varietà di versioni per diversi gusti audio, include anche pulsanti fisici, un telecomando e puoi persino ottenere un case specifico per esso. Un paio di modelli hanno anche un piccolo schermo per visualizzare cosa stai ascoltando!

- Da 42€ / 48$
- magpi.cc/XdxPxP

HiFiBerry DAC+

I DAC HiFiBerry sono molto popolari, e disponibili in una gamma di taglie e modelli diversi per qualsiasi budget, requisiti audio o tipo Raspberry Pi. C'è anche una versione con uscita digitale/SPDIF.

- Da 22€ / 26$
- magpi.cc/aimAyy







Nanosound Player

Questo kit tutto-in-uno utilizza il Nanosound AMP per creare un sistema stereo completo a cui devi solo collegare gli altoparlanti. Viene fornito anche il telecomando dell'altro DAC e AMP, ed è tutto in uno case ben rifinito.

- 146€ / 169$
- magpi.cc/tCRnde

Pi-DigiAMP+

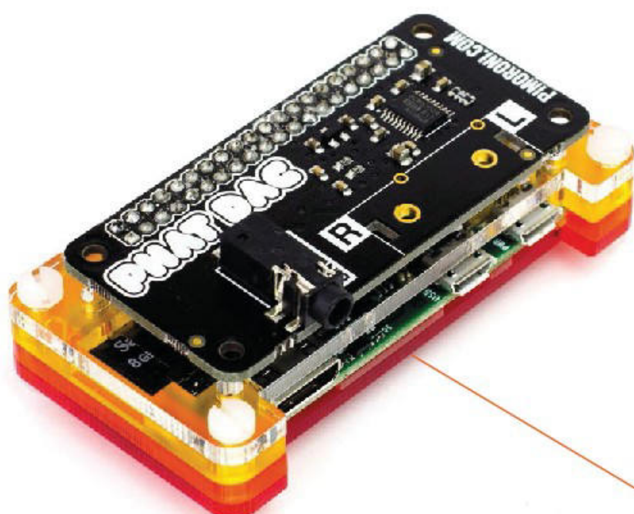
IQAudIO è uno dei marchi che produce add-on audio per il Raspberry Pi e la sua gamma DigiAMP è stata recentemente aggiornata Sebbene fosse già un eccellente prodotto.

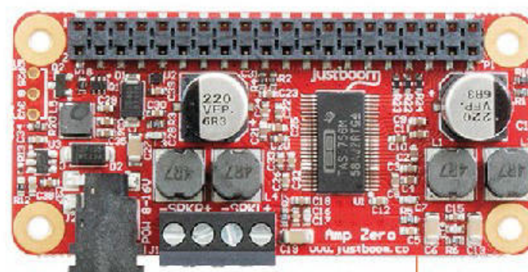
- 60€ / 69$
- magpi.cc/fisaLH











Pi-DACZero

Questo DAC delle dimensioni dello Zero è il fratello del DAC audio di dimensioni standard per gli altri modelli di Pi. Considerando che il Pi Zero non ha una uscita audio, avrai bisogno di qualcosa di simile per quasi tutte le soluzioni in cui ti serva che il Pi Zero emetta dei suoni.

- 19€ / 21$
- **magpi.cc/sxVfhU**

pHAT DAC

Questo DAC carino ed economico aggiunge una uscita audio da 3,5 mm - vale a dire un jack per cuffie - al Pi Zero. Che è ben fatto, ma normalmente non ne ha una. Puoi anche saldarci sopra delle porte RCA, per espanderlo.

- 13€ / 13$
- **magpi.cc/SUsJrP**

Amp Zero pHAT

Un amplificatore per il Pi Zero? Assolutamente, e questo è uno dei pochi che abbiamo trovato. JustBoom produce una grande gamma di amplificatori e DAC e questo non fa eccezione: è solo molto più piccolo.

- 27€ / 37$
- **magpi.cc/2iNJMKP**



Speaker pHAT

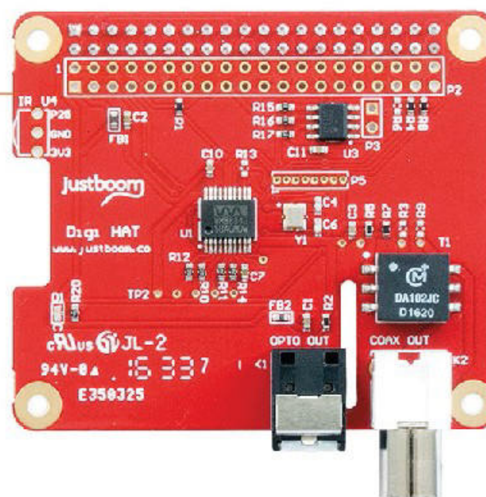
Questo DAC va oltre la possibilità di iniettare musica in uno stereo – rende il Pi Zero uno stereo, grazie al suo piccolo altoparlante incorporato. Ha anche un indicatore grafico del volume per mostrare quanto forti sono le tue melodie.

- 13€ / 13$
- **magpi.cc/2kXdZsE**

Digi HAT

Con uscite ottica e coassiale, potrai ottenere un po' di musica in alta qualità con il Digi HAT. JustBoom vende anche, se preferisci, versioni di DAC tradizionali con prese RCA standard.

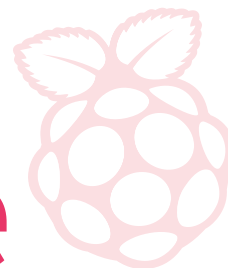
- 33€ / 43€
- **magpi.cc/2lMEYFx**





PLAYER MUSICALI

Con l'ultima versione di Raspbian, il lettore multimediale predefinito su Raspberry Pi è ora VLC. Gestisce molti tipi di codec ed è altamente configurabile, rendendolo il software perfetto per il tuo Pi con DAC.



Questo mese in Raspberry Pi

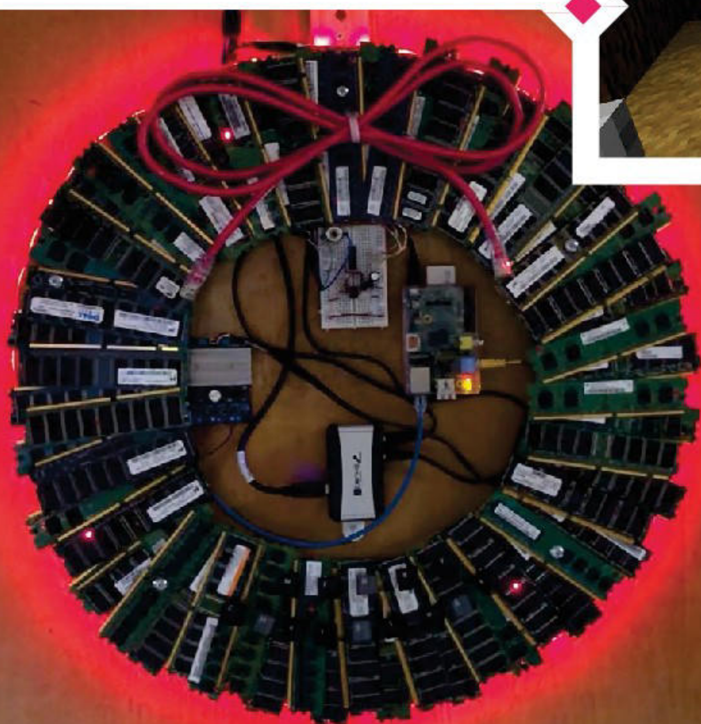
Natale 2018

Buone feste dalla terra del Raspberry Pi!

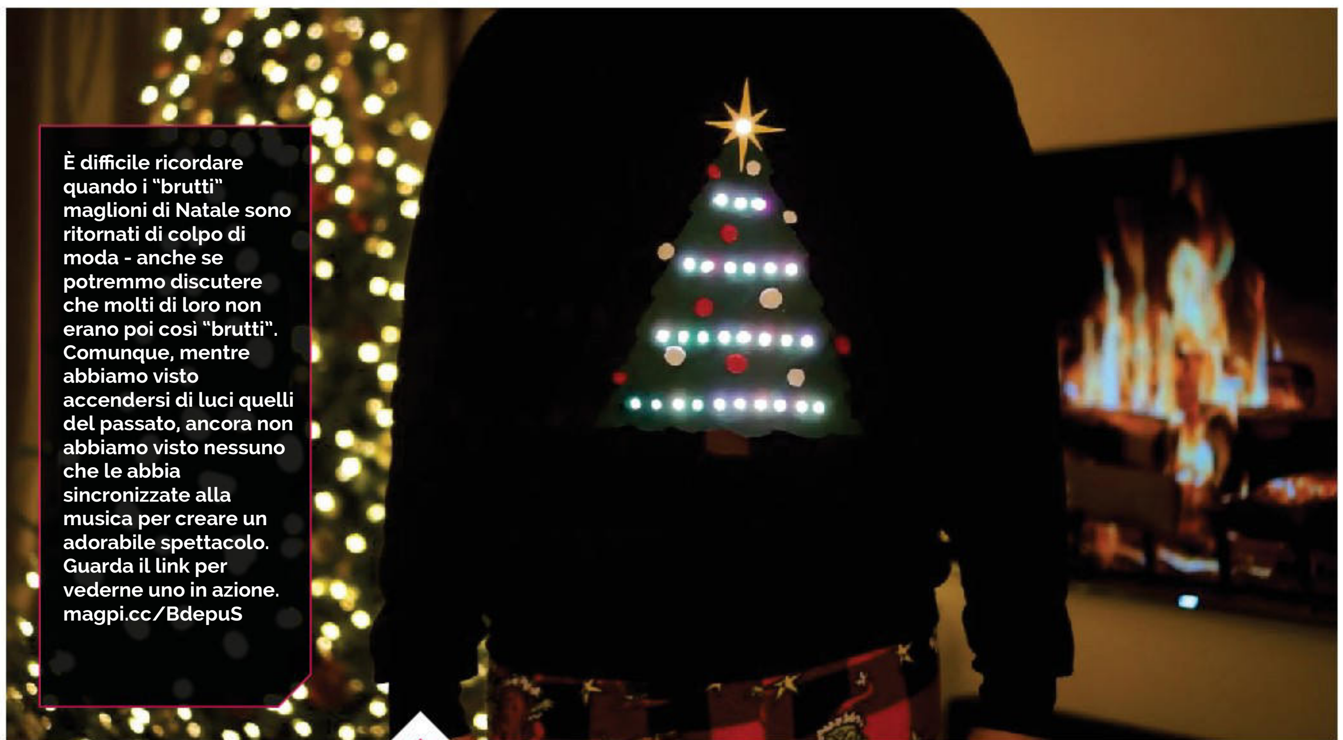
Anche se questo è il nostro numero di Gennaio, probabilmente lo leggerai ancora nel periodo delle feste. Quindi per rimanere fugacemente in tema con la stagione, ecco alcuni grandi progetti Pi di Natale che abbiamo visto questo mese!



Utilizzando Minecraft Pi, i ragazzi di Brocraft Gaming stanno controllando un vero albero di Natale. Puoi interagirci visitando il loro server su Minecraft: Edizione Java per PC e Mac, raccogli power, e parla con un NPC sotto un albero nel gioco per cambiare i giochi di luci su entrambi gli alberi. Accertati di avere il live streaming attivo, così puoi vedere i cambiamenti che fai! magpi.cc/VBxjcQ



Cosa fai quando non hai una ghirlanda per Natale? Pare che un ufficio abbia deciso di farne una con stecche di RAM e un po' di luci a LED dietro di esse, alimentata da un Raspberry Pi. Come potresti fare tu. Ci piace molto il cavo Ethernet legato a fiocco nella parte superiore. Ed è davvero in vendita. magpi.cc/jjdWiz



È difficile ricordare quando i "brutti" maglioni di Natale sono ritornati di colpo di moda - anche se potremmo discutere che molti di loro non erano poi così "brutti". Comunque, mentre abbiamo visto accendersi di luci quelli del passato, ancora non abbiamo visto nessuno che le abbia sincronizzate alla musica per creare un adorabile spettacolo. Guarda il link per vederne uno in azione. magpi.cc/BdepuS



Questa decorazione per l'ufficio riproduce spot pubblicitari casuali di Natale e festività, che puoi mischiare con la semplice pressione di un pulsante. Ovviamente riproduce anche i trailer dei classici di Natale come The Muppet Christmas Carol. Speriamo che abbia trailer per altri preferiti come Arma Letale, Kiss Kiss Bang Bang, Iron Man 3, e altri film della filmografia del re di Natale stesso, Shane Black. magpi.cc/DzENuQ



Questa folta schiera di luci è controllata da un Raspberry Pi e due microcontrollori. È in continua evoluzione e cambiano i colori, quindi è magnifico da guardare in movimento - guarda un video nel link in basso! Abbiamo alcune luci di Natale alimentate da Pi, ma niente di più ridicolo, in confronto a questo. magpi.cc/TdhEFr